



**University of
Zurich**^{UZH}

**Zurich Open Repository and
Archive**

University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2011

Digging for names in the mountains: Combined person name recognition and reference resolution for German alpine texts

Ebling, S ; Sennrich, R ; Klaper, D ; Volk, Martin

Abstract: In this paper we introduce a module that combines person name recognition and reference resolution for German. Our data consisted of a corpus of Alpine texts. This text type poses special challenges because of a multitude of toponyms, some of which interfere with person names. Our reference resolution algorithm outputs person entities based on their last names and first names along with their associated features (jobs, addresses, academic titles).

DOI: https://doi.org/10.1007/978-3-319-08958-4_16

Posted at the Zurich Open Repository and Archive, University of Zurich

ZORA URL: <https://doi.org/10.5167/uzh-50451>

Conference or Workshop Item

Accepted Version

Originally published at:

Ebling, S; Sennrich, R; Klaper, D; Volk, Martin (2011). Digging for names in the mountains: Combined person name recognition and reference resolution for German alpine texts. In: 5th Language Technology Conference, Poznan, Poland, 25 November 2011 - 27 November 2011.

DOI: https://doi.org/10.1007/978-3-319-08958-4_16

Digging for Names in the Mountains: Combined Person Name Recognition and Reference Resolution for German Alpine Texts

Sarah Ebling, Rico Sennrich, David Klaper, Martin Volk

Institute of Computational Linguistics, University of Zurich
Binzmühlestrasse 14, 8050 Zurich, Switzerland
{ebling,sennrich,volk}@ifi.uzh.ch, david.klaper@access.uzh.ch

Abstract

In this paper we introduce a module that combines person name recognition and reference resolution for German. Our data consisted of a corpus of Alpine texts. This text type poses special challenges because of a multitude of toponyms, some of which interfere with person names. Our reference resolution algorithm outputs person entities based on their last names and first names along with their associated features (jobs, addresses, academic titles).

Keywords: named entity recognition, person name recognition, reference resolution, German

1. Introduction

Named entity recognition (NER) is a prerequisite for many language technology applications, such as Information Extraction or Question Answering. Commonly it involves identifying people, companies, and locations. The difficulty of NER depends heavily on the language under consideration. For English, it is often sufficient to look for non-sentence-initial capitalized words. In particular, due to the restricted word order of English, a capitalized word preceding a verb of communication has a high likelihood of being a person name or organization name (Rössler, 2004).

For other languages such as German, NER is more challenging. In German, both regular nouns and proper names are capitalized, with only the latter being candidates for named entities (NEs). This means that by itself, capitalization is not a viable indicator of German NEs. Word order is not a stand-alone predictive clue, either, since German allows for multiple positions of the finite verb and the subject.

The task of detecting names of people, known as person name recognition (PNR), is a subtask of NER. In this paper we describe a combined approach to PNR and reference resolution for German. Our data consisted of a corpus of Alpine texts. This text type poses special challenges because of a multitude of toponyms, some of which interfere with person names. The remainder of this paper is structured as follows: In Section 2 we give an overview of the approaches that have been pursued to tackle PNR for German. In Section 3 we describe our approach to this issue and report the results of our experiments. We also outline our take on reference resolution for person entities.

2. Person Name Recognition for German

Volk and Clematide (2001) pursued a “learn—apply—forget” approach to detecting person names, geographic names, and company names in a computer magazine corpus. They based their approach on the assumption that last names do not occur unaccompanied when they are introduced but instead appear along with a first name or another predictive marker. The authors used a list of

16,000 first names derived from electronically available telephone directories as well as a handcrafted list of titles and jobs. Their system accepts last names retrieved in this manner in their stand-alone form for a predefined text span (e.g., 15 sentences), after which it “forgets” them. The authors reported a precision of 92% when testing their approach on a set of 990 German sentences. Recall was 93% for full names, and 74% for stand-alone last names.

Florian et al. (2003) were among the participants of the CoNLL 2003 shared task on “Language-Independent Named Entity Recognition” (Tjong Kim Sang and De Meulder, 2003). This task produced what is to date the only freely available data for German NER, a collection of articles from the newspaper “Frankfurter Rundschau” annotated with four NE categories: person, location, organization, and miscellaneous (Faruqui and Pado, 2010). The training set consists of 220,000 tokens, and the development and test set of 55,000 tokens each. Florian et al. combined four classifiers: a robust linear, a maximum entropy, a transformation-based learning, and a hidden Markov model classifier. Among the features they used were words and their lemmas in a 5-word-window relative to the word under consideration, part-of-speech tags, typographical cues, and the output of two other NE classifiers. For German they additionally experimented with lists of first names, last names, place names, and country names. They observed an increase in performance when adding the lists to their classifiers, obtaining results of up to 91.93% precision, 75.31% recall, and 82.80% f-measure.

Rössler (2004) used the same data set as Florian et al. (2003) but focused on the person category exclusively. He applied a linear SVM classifier that relied on context features and word-internal features, e.g., morphological and typographical cues. In addition to this baseline classifier he used a corpus lexicon that recorded the frequency of a word conditioned on its appearance as a person entity. He obtained the lexicon by training a weak SVM classifier on the CoNLL training data and applying it to a 40-million word corpus. The output consisted of 320,000 word forms tagged as potential person names along with a confidence value assigned by the classifier. Rössler observed performance gains when including the corpus

lexicon. His approach evaluated to a precision of 89.4%, a recall of 88.4%, and an f-measure of 88.9%.

Faruqui and Pado (2010) applied clustering based on distributional and morphological similarity to unlabeled data in order to obtain classes of words that belong to the same NE category. They included morphological similarity because distributional similarity by itself leads to unreliable results for infrequent words. By performing morphological analysis, their system was able to recognize infrequent words like *Ostdeutschland* ('East Germany') and *Westdeutschland* ('West Germany') as similar to *Deutschland* ('Germany') and assign them the same NE category tag. The authors experimented with different corpora as their unlabeled data and found that the Huge German Corpus (175M tokens of newspaper text) yielded the best results. They used the Stanford Named Entity Recognizer, which allows for the inclusion of similarity features and, apart from these, considers words, lemmas, and part-of-speech tags. The results for the person category on the CoNLL 2003 shared task test set were 96.2% (precision), 88.0% (recall), and 92.0 (f-measure). Performance dropped by approximately 10% when the authors applied their system to a set of German sentences from the out-of-domain Europarl corpus.

3. PNR and Reference Resolution

While the above papers offer ample ways to solve the problem of identifying person names in German texts, none of them addresses the issue of reference resolution that needs to succeed PNR. Reference resolution refers to the task of "determining what entities are referred to by which linguistic expressions" (Jurafsky and Martin, 2009, p. 729). Ideally, annotations below the person name level should be available for this, i.e., at the least, a distinction between first names and last names should exist. This is not the case with the CoNLL 2003 shared task data.

Our motivation to tackle reference resolution for person entities arose from the practical need to provide such output within our *Text+Bern* project (Bubenhofer et al., 2011).¹ The aim of this project is to compile a multilingual heritage corpus of Alpine texts from different sources that can be used, e.g., as data for domain-specific machine translation (Sennrich, 2011). Earlier this year, we released the complete range of yearbooks of the Swiss Alpine Club from its start in 1864 to today. The yearbooks contain reports on mountain expeditions as well as information about the geology, flora, and fauna of the Swiss mountains. Some of the books are multilingual, with articles written in German, French, Italian, Romansh, and Swiss German; others are available in two monolingual volumes, a German and a French one. In total they amount to 87,000 pages in 196 volumes. This corresponds to 35.75 million word tokens.

We are currently adding more annotation layers to the corpus. An important part of this effort is the recognition of person names in the German part of the corpus. The German part makes up 61% of the total corpus size. In what follows, we describe our work in this area. We introduce our reference resolution algorithm in Section 3.2.

3.1. Person Name Recognition

Our PNR algorithm is a decision tree with handcrafted rules. It considers stand-alone first names, stand-alone last names, and combinations thereof. Candidates for last names are strings that start with a capitalized letter and are at least three characters long. As special characters they may contain hyphens. One or more titles of nobility may immediately precede a last name core so as to capture occurrences like *van der Vaart*, *von Muralt*, etc. In order to qualify as a last name, a candidate must meet at least one of the following conditions (cf. Fig. 1):²

1. **First name complex:** the last name candidate appears after an arbitrary number of full first names (FFN) and/or abbreviated first names (AFN). Optionally, a predictive word complex (cf. 2.) may precede a first name complex.
2. **Predictive word complex:** the last name candidate appears after a predictive word complex, which is made up of a job marker (e.g., *Förster*, 'ranger'), an address marker (e.g., *Herr*, 'Mister'), an academic title marker (e.g., *Dr.*, 'Dr.'), or a combination thereof (e.g., *Herr Förster*, *Herr Dr.*, etc.). Address markers convey gender information that is carried on to the reference resolution step (cf. Section 3.2). The order within the predictive word complex is variable, and markers may appear more than once (e.g., *Prof. Dr. Dr.*). First name complexes may not precede predictive word complexes, as a sequence like *Joachim Herr Förster* would not be in accord with the natural naming order.

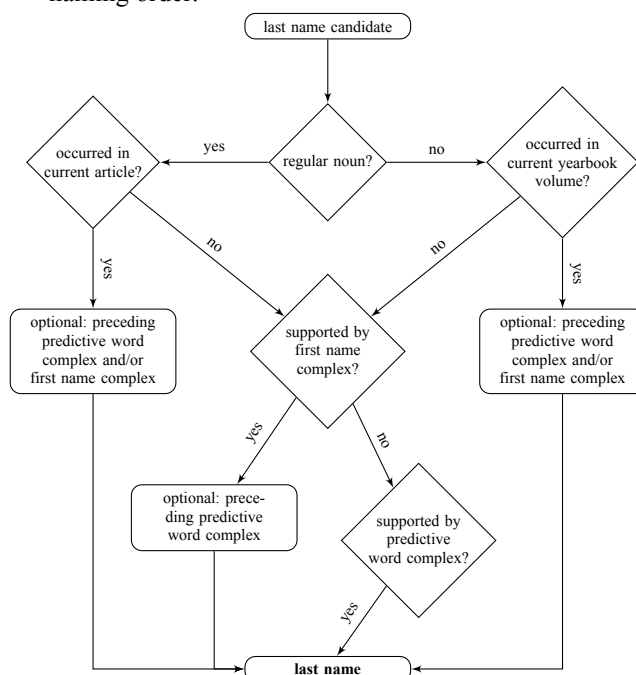


Fig. 1: Deciding whether a candidate is a last name

3. **Recurrence:** a previous occurrence of the last name candidate was marked as a last name in the current yearbook volume. Recall that regular nouns in German start with capital letters; hence, the task of distinguishing a person name from a regular noun is not

¹ <http://www.textberg.ch/>

² The first two restrictions build on the assumption previously made by Volk and Clematide (2001) that last names are generally not introduced in isolated form.

trivial. If a person name is also a regular noun (which is a common kind of ambiguity in our corpus³), the priming span is reduced and the last name candidate only considered if a previous occurrence was marked as a last name in the current article. We determine whether a candidate is a regular noun by consulting a list of 394,775 regular nouns. A first name complex and/or predictive word complex may optionally precede a recurrent last name.

We prevent a number of last names from being recognized; most of them are toponyms, e.g., *Altmann*, the name of a mountain. In the following two sections we discuss the first name complex and the predictive word complex in more detail.

3.1.1. First Name Complex

An abbreviated first name (AFN) starts with a capitalized letter and ends with a period mark. Apart from that it may contain an arbitrary number of lower case letters. Not restricting this number was motivated by the observation that AFN are rather long in our corpus (e.g., *Joach.* is used as an abbreviation for *Joachim*). We blocked a number of abbreviations, such as *Gd.* (for *Gemeinde*, ‘community’), or *Nr.* (for *Nummer*, ‘number’). We identified full first names based on a list of 58,969 first names along with their genders obtained from the web. The list does not feature complex first names such as *Jean-Pierre* or *Franz-Sepp*. We split such occurrences and matched their individual parts against the list, assigning the gender of the first part to the entire first name. Both for simple and for complex first names we only assigned the gender feature if it was not ambiguous (a first name like *Andrea* is ambiguous as it can be both male and female). We manually removed first names that had a high potential of ambiguity—such as *Tod* (‘death’), *Lücke* (‘gap’), or *Birke* (‘birch tree’)—from the list.

3.1.2. Predictive Word Complex

Our observation that job markers are an important indicator of person names led us to look for a comprehensive list of jobs. We found the Swiss Standard Classification of Occupations 2000⁴, which is the currently valid classification scheme for demographic surveys. The list comprises 19,000 entries. We performed a number of alterations, including:

- Manually removing entries with a high risk of ambiguity, e.g., *Faktor*, which can be a near-synonym of ‘criterion’ and ‘commissioner’, with only the second denoting a job title.
- Manually marking entries that are well suited to retrieve person names but are not job titles in a strict sense, e.g., *Retter* (‘rescuer’), or *Spezialist* (‘specialist’). We called them *restricted job markers*.
- Automatically extending the list by adding old spelling variants. We applied the following orthographic rules: $t \rightarrow th$, $z \rightarrow c$, $k \rightarrow c$, $ier \rightarrow ir$. They led to additional job hits like *Karthograph* (‘cartog-

rapher’, modern spelling *Kartograph*), or *Diacon* (‘deacon’, modern spelling *Diakon*).

- Automatically extending the list by adding morphological variants

During our preliminary experiments we observed that two co-occurrence patterns containing job titles produced many errors: firstly, the pattern *title of nobility+job title* yielded false hits like *Jäger von Pontresina* (‘hunterman from Pontresina’, where our system correctly identified ‘hunterman’ as a job title and from there inferred that ‘von Pontresina’ was a last name containing a title of nobility), or *Führer von Zermatt* (‘mountain guide from Zermatt’). Secondly, the co-occurrence of a job title and a last name candidate that was also a regular noun accounted for false last names like *Bergsport* in *Sachbearbeiter Bergsport* (‘administrative assistant mountaineering’). We therefore decided to disable these two patterns.

3.2. Reference Resolution

Our intended output was a collection of person entities for each yearbook. In practice, this meant that we had to aggregate all features (address, job, academic title) that were available for an entity. We used last names as first-level discriminators, and first names (both full first names and abbreviated first names) as second-level discriminators for entities. In other words, we introduced a new entity for every distinct FFN or AFN associated with a particular last name. Where there were only stand-alone occurrences of a last name (i.e., no FFN or AFN), we merged all features and produced one single entity. Where there were stand-alone occurrences as well as one distinct FFN or AFN, we did the same.⁵ As an example, assume that a last name *Ferrari* occurred both stand-alone as well as in combination with one first name, *Casimiro*. We aggregated the features of all occurrences and created a single entity *Casimiro Ferrari*.

The procedure got fairly complex in cases where there were multiple FFN or AFN or combinations of both along with stand-alone occurrences of a last name. For example, if a last name *Fischer* appeared by itself as well as in conjunction with the first names *Scott* and *Miriam*, we produced three new entities—*Fischer*, *Scott Fischer*, and *Miriam Fischer*—, since it was not clear which of the two entities specified with their first names the stand-alone last name referred to. We applied the same strategy on a lower level: if for an AFN there was only one FFN with the same initial letter (e.g., AFN *A.*, FFN *Aegidius*), we considered them to refer to the same entity; otherwise (e.g., AFN *A.*, FFN *Aegidius* and *Arnold*) we refrained from merging. We also merged different AFN, e.g., *Osw.* and *O.*

We allowed for multiple job markers to be output for one entity. This case appeared frequently, with one title often being more specific than the other, e.g., *Bergführer* and *Führer* (‘mountaineering guide’ and ‘guide’). Note that the more general job title (*Führer*) is still more specific than any of the restricted job markers we mentioned

³ Our corpus contains person name instances like *Peter Eimer* (‘Peter Bucket’), *Paul König* (‘Paul King’), *Herr Sand* (‘Mister Sand’), or *M. Held* (‘M. Hero’).

⁴ http://www.bfs.admin.ch/bfs/portal/en/index/infothek/nomenklaturen/blank/blank/sbn_2000/01.html

⁵ Note that both combining multiple stand-alone occurrences into a single entity and merging stand-alone occurrences with distinct FFNs or AFNs is potentially erroneous. We pursued this procedure to obtain a consolidated collection of person entities.

in Section 3.1.2. We did not consider restricted job markers in the output.

Below is an example of the output for two entities that belong to the same last name, *Studer*. It summarizes various aspects of our reference resolution algorithm: entities may carry academic title markers (*Prof.*), job markers (*Geologe*, ‘geologist’), and address markers (*Herr*, ‘Mister’), AFN and FFN are aggregated (*B. Bernhard* and *G. Gottlieb*, respectively), and entities receive a gender feature based on gender-annotated first names (*Bernhard*, male) or address markers (*Herr*, male).

```
<person id="83">
  <address></address>
  <titles>Prof.</titles>
  <firstname>B. Bernhard</firstname>
  <lastname>Studer</lastname>
  <gender>M</gender>
  <profession>Geologe</profession>
  <references>
    <reference>
      <positions>
        <position>32-18-17</position>
        <position>32-18-18</position>
        <position>32-18-19</position>
        <position>32-18-20</position>
      </positions>
    </reference>
  </references>
</person>
<person id="84">
  <address>Herr</address>
  <titles></titles>
  <firstname>G. Gottlieb</firstname>
  <lastname>Studer</lastname>
  <gender>M</gender>
  <profession></profession>
  <references>
    <reference>
      <positions>
        <position>37-54-23</position>
        <position>37-54-24</position>
        <position>37-54-25</position>
      </positions>
    </reference>
  </references>
</person>
```

3.3. Evaluation

To evaluate our PNR algorithm (Section 3.1) we created a gold standard annotation for the German parts of two of our Alpine yearbook volumes, 1901 (development set, 1933 person name tokens) and 1994 (test set, 978 person name tokens). We annotated outer-level entities only, meaning that we did not mark person names within, e.g., toponyms as named entities (cf. Section 3.3.2 for examples). We evaluated on two levels: the single-token level, and the entity level (cf. Table 1). On the entity level, an entity was a true positive match only if all of its parts were annotated as person name tokens (e.g., for *Horace Benedict de Saussure* to become a true positive entity, all four tokens, including the title of nobility *de*, had to be marked as person names).

3.3.1. Finding the Optimal Configuration

We performed a number of experiments on the development set to determine which were necessary restrictions to impose on our person name recognition algorithm. Among the configurations we tested were:

1. Allowing for a last name recognized via an AFN to be a regular noun, e.g., *Eimer* in *P. Eimer* (‘P. Eimer’ vs. ‘P. Bucket’)
2. Allowing for a title of nobility and a job marker to co-occur, e.g., *Präsident von Planta* (cf. Section 3.1.2)
3. Permitting AFN from our list of blocked abbreviations, e.g., *St.*, which can be both short for ‘Saint’ and for the first name ‘Stefan’ (cf. Section 3.1.1)
4. Permitting last names from our list of blocked last names, e.g., *Altmann* (cf. Section 3.1)
5. Allowing for a job marker and a regular noun to co-occur, e.g., in the pattern *Sachbearbeiter Bergsport* (cf. Section 3.1.2)

Our initial setting was a combination of the options marked with a star in Table 1, where “yes” means that the features of the respective configuration were considered, and “no” means that they were not considered. In other words, in our initial setting, we allowed for a last name recognized via an AFN to be a regular noun (configuration 1), while we did not allow for a title of nobility and a job marker to co-occur (2), we did not permit AFN from our list of blocked abbreviations (3), nor last names from our list of blocked last names (4), and we did not allow for a job marker and a regular noun to co-occur (5).

Table 1 displays the results of our experiments on the development set in terms of recall (R), precision (P), and f-measure (F). We opted for high precision. The better of the two options (“yes” vs. “no”) for each of configurations 1 to 5 with respect to precision is printed in bold. We found that the better-performing options were the ones marked with a star; hence, our initial setting (options “yes”, “no”, “no”, “no”, “no”) performed best. In particular, we gained evidence that blocking the co-occurrence of an AFN and a last name that is a regular noun (option “no” for configuration 1) did not aid precision.

		single-token level			entity level		
		P	R	F	P	R	F
1	yes*	84.38	62.34	71.70	69.07	54.43	60.88
	no	83.98	61.82	71.22	66.84	53.70	59.55
2	yes	83.29	63.17	71.84	68.06	55.73	61.28
	no*	84.38	62.34	71.70	69.07	54.43	60.88
3	yes	77.34	64.84	70.54	63.64	56.84	60.05
	no*	84.38	62.34	71.70	69.07	54.43	60.88
4	yes	82.55	63.34	71.68	67.52	55.80	61.10
	no*	84.38	62.34	71.70	69.07	54.43	60.88
5	yes	83.57	63.17	71.95	68.39	55.73	61.41
	no*	84.38	62.34	71.70	69.07	54.43	60.88

Table 1. Recall (R), precision (P), and f-measure (F) for the different configurations on the development set (in %)

3.3.2. Evaluating on the Test Set

On the test set we applied the options printed in bold in Table 1. For the single-token level, this yielded a precision of 87.34%, a recall of 77.61%, and an f-measure of 82.19%. For the entity level, the corresponding numbers are 75.47% (precision), 71.22% (recall), and 73.28% (f-measure). To get an idea of which of the three person name classes—full first names, abbreviated first names, and last names—was the most difficult to recognize, we performed a more fine-grained evaluation. The results for

each class are displayed in Table 2.⁶ They show that in terms of precision, last names were the easiest class to recognize, while full first names were the hardest. Paired with the observation that full first names had the highest recall this suggests that the full first names in our list still have a high potential of ambiguity. At the same time the low recall score of last names and abbreviated first names indicates that the restrictions we imposed on these two classes, presumably via the lists of blocked last names and blocked abbreviations, were too severe.

	True pos.	True neg.	False pos.	False neg.	P	R	F
all	759	68566	110	219	0.8734	0.7761	0.8219
LN	298	68566	24	185	0.9255	0.6170	0.7404
FFN	398	68566	124	56	0.7625	0.8767	0.8156
AFN	16	68566	2	14	0.8889	0.5333	0.6667

Table 2. Recall (R), precision (P), and f-measure (F) for the test set relative to last names (LN), full first names (FFN), and abbreviated first names (AFN)

When examining the false-positive instances in our overall experiment (row “all” in Table 2) we found one major source of errors: person entities nested within other entities, especially person entities embedded in toponyms. Examples include the first name *Luigi* and the last name *Amadeo* in *Pizzo Luigi Amadeo* (‘Luigi Amadeo Peak’). We had refrained from annotating such occurrences since we had only considered outer-level person entities (cf. Section 3.3). We found that of 110 false positives, 54 were due to this phenomenon, i.e., were legitimate person name tokens on the inner level.

We also observed that the one-sense-per-document hypothesis did not hold true for our corpus: we found instances of (outer-level) person entities whose components later appeared as part of geographic entities. For example, the phrase *Der italienische Alpinist Mannelli* (‘The Italian mountaineer Mannelli’) preceded appearances of the last name *Mannelli* in the toponyms *Couloir Mannelli* (‘Mannelli Channel’), and *Capanna Mannelli* (‘Mannelli Cabin’). Similarly, the person name *Heinrich Harrer* occurred, and the last name *Harrer* later appeared as part of the toponym *Harrer Lake*. Volk and Clematide (2001) made a similar observation for person names and company names. Upon inspecting the false-negative instances, we found that last names often occurred both in their nominative and in their genitive or plural form (-s). For example, our system recognized the person name *John Tyndall* (first name *John*, last name *Tyndall*) but failed to identify the genitive form of the last name, *Tyndalls*, that appeared stand-alone at a later stage.

4. Conclusion

In this paper, we reported our work on person name recognition. Our data consisted of German Alpine texts. We considered stand-alone first names and stand-alone last names as well as combinations thereof. We started with the assumption that last names are not introduced

unaccompanied but instead occur in conjunction with either a first name complex or a predictive word complex at their first appearance. Predictive words are job markers, address markers, and academic title markers. We also performed reference resolution. We took into account the fact that a first name can be cited in full form or in various abbreviated forms. We also considered the fact that last names that occur stand-alone as well as in conjunction with a single first name are likely to belong to the same entity.

As our next step, we plan to augment our PNR algorithm by considering the genitive/plural forms of introduced last names. We will also experiment with coordinated person names (‘the mountaineers Müller and Fischer’), and predictive words following person names (‘Müller, a mountaineer, ...’). Further we plan to improve our PNR system for German by drawing on information from the French part of our corpus. We will exploit the fact that regular nouns in French are not capitalized, and we will disambiguate noun-name ambiguities in parallel texts based on co-occurrence vs. translation in both versions. For example, the German word *Zweifel* is only a last name if it also occurs in the French parallel text; if it is translated as *doute*, then its reading as last name can be discarded. Our system for German PNR and reference resolution is written as a Python module and will be made available upon request.

References

- Bubenhof, N., Volk, M., Althaus, A., Jitca, M., Bangerter, M., Sennrich, R. (eds.) (2011): Text+Berg-Korpus (Release 145). Institut für Computerlinguistik, Universität Zürich.
- Faruqui, M. and Padó, S. (2010). Training and Evaluating a German Named Entity Recognizer with Semantic Generalization. In *Proceedings of KONVENS 2010*, Saarbrücken, Germany.
- Finkel, J.R. and Manning, C.D. (2009). Nested named entity recognition. In *Proceedings of EMNLP*, pp. 141-150, Singapore.
- Florian, R., Ittycheriah, A., Jing, H. and Zhang, T. (2003). Named entity recognition through classifier combination. In *Proceedings of CoNLL 2003*, pp. 168-171, Edmonton, Canada.
- Jurafsky, D. and Martin, J. H. (2009). *Speech and Language Processing* (2nd ed.). Upper Saddle River, New Jersey: Pearson Education International.
- Rössler, M. (2004). Corpus-based Learning of Lexical Resources for German Named Entity Recognition. In *Proceedings of LREC 2004*, Lisbon, Portugal.
- Sennrich, R. (2011). Combining multi-engine machine translation and online learning through dynamic phrase tables. In: *Proceedings of EAMT 2011*, pp. 89-96, Leuven, Belgium.
- Tjong Kim Sang, E.F. and De Meulder, F. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of CoNLL 2003*, pp. 142-147, Edmonton, Canada.
- Volk, M. and Clematide, S. (2001). Learn – filter – apply – forget. Mixed approaches to named entity recognition. In *6th International Workshop on Applications of Natural Language for Informations Systems*, Madrid, Spain.

⁶ We did not compare our approach with other PNR systems because we aimed for a more fine-grained classification.